



Adaptive Dual Color Visible Light Communication (VLC) System

Antonio Costanzo, Valeria Loscrì, Sandra Costanzo

► To cite this version:

Antonio Costanzo, Valeria Loscrì, Sandra Costanzo. Adaptive Dual Color Visible Light Communication (VLC) System. 6th World Conference on Information Systems and Technologies, Mar 2018, Napoli, Italy. hal-01687543

HAL Id: hal-01687543

<https://inria.hal.science/hal-01687543>

Submitted on 18 Jan 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Adaptive Dual Color Visible Light Communication (VLC) System

Antonio Costanzo¹, Valeria Loscri¹, and Sandra Costanzo²

¹ Institute National de Recherche en Informatique et en Automatique, INRIA
Lille-Nord Europe, Villeneuve d'Ascq, 59650, France
`antonio.costanzo@inria.fr`, `valeria.loscri@inria.fr`

² Dipartimento di Ingegneria Informatica, Modellistica, Elettronica e Sistemistica,
DIMES, Università della Calabria, Rende, 87036, Italy
`costanzo@dimès.unical.it`

Abstract. In this work, we propose a Visible Light Communication (VLC) system dynamically adapting, through a decision-making process based on a simple fuzzy-logic, the transmitting color selection to the external environmental conditions. Transmitted Signals are opportunistically treated through softwarization approaches by using basic hardware (i.e. Arduino boards and inexpensive LEDs in the transmitting stage) in order to implement an effective, end-to-end, adaptive communication system. In particular, we will show that, if a low environmental noise is added, the system keeps to be well-performing in terms of Bit Error Rate (BER) also at higher distances (up to 8-9 meters) using a warm white front-end, while, if high external interfering lights are present in the environment, a low power red front end is dynamically fed for maintaining a good-level communication (with low Bit Error Rate).

Keywords: Visible Light Communication, Signal Processing, Adaptive Systems

1 Introduction

Visible Light Communication (VLC) is a short range optical wireless technology based on LEDs for illumination and communication purpose simultaneously [1, 2]. The integration of VLC paradigm in existing infrastructures represents a very powerful tool to boost diffusion of novel technologies [3], coexisting and inter-operating with others EM devices based on different paradigms [4]. As compared with "traditional" radio-frequency based communication systems, VLC gives several potential advantages. Apart from the possibility to use existing lighting infrastructures for both illumination and data communication, VLC can contribute to the development of novel high performance geo-localization systems within indoor environments, where GPS based techniques are not affordable, as well as in the framework of potential pervasive networks in hospitals, aircraft and other areas where the interference of classical RF devices with a critical instrumentation could represent a serious hazard.

Due to recent improvements in optical devices, allowing high speed of the transmitting and receiving front-ends, most of modulation schemes proposed in literature for visible light applications face the main goal of data rate maximization. So, modified On Off Keying (OOK), Orthogonal Frequency Division Multiplexing (OFDM) and, in the latter years, Color Shift keying are mainly used [6]. However, in order to really make VLC a pervasive paradigm, the simplicity in its realization while keeping robustness, and reliability feature are fundamental characteristics to be considered and integrated in the system design. Bearing that in mind, we designed and developed a VLC system based on On Off Binary Frequency Shift Keying, a modulation technique normally disregarded in modern communication systems due to its spectral inefficiency, but allowing a very simple data encoding and decoding mechanism, available for good performance in a highly noisy environment even using extremely basic hardware devices [5]. The overall signal operations in our system (i.e. transmission coding, receiver decoding, FSK modulation, filtering, etc.), except from the led driver in the transmitting stage and the trans-impedance amplifier in the receiving stage, have been implemented via software, in order to make the system as much flexible and low cost as possible, following a Software Defined paradigm, widely used in current literature in the framework of radio and radar systems [7] and [8]. Since all VLC system performances extremely depend on external light conditions, in order to properly face with environmental luminosity changes (e.g. interfering artificial lights not belonging to the communication system), an adaptive scheme for maintaining a minimum Quality of Service (QoS) level of communication between transmitter and receiver is necessary. In this work, we refer to QoS as a Bit Error Rate (BER) threshold value. Cold white front-end is considered as primary source in our communication system, since it is also used for illumination purpose, but we also consider a red front-end (with no illumination aim) that behaves better in noisy environments.

This aspect could be interesting for several indoor applications, such as hospitals, where sunlight and artificial lights, not being part of the VLC system, could significantly prevent its usage for long periods, or normal office rooms, where the standard warm white VLC system could be turn off without losing the communication when the natural light is enough strong for properly lighting without the use of other lights. The rest of the paper is organized as follows. In Section 2 we describe the architecture of our system and the signal processing. Section 3 describes the experiments we have dealt to validate the proposed approach. Finally, we conclude the paper in Section 4, where we give some future insights.

2 System Description and Signal Processing

A simplified architecture of our system, describing both transmitting and receiving devices is depicted in Figure 1, while a block diagram including the main operations allowing the color switching algorithm in the transmitting path is provided in Figure 2.

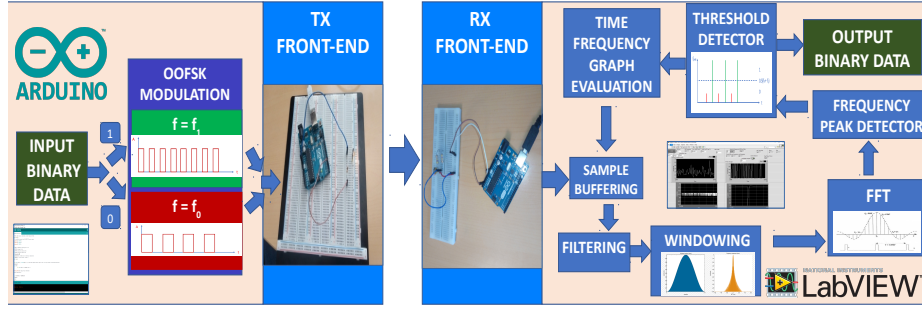


Fig. 1. Architecture of the VLC system.

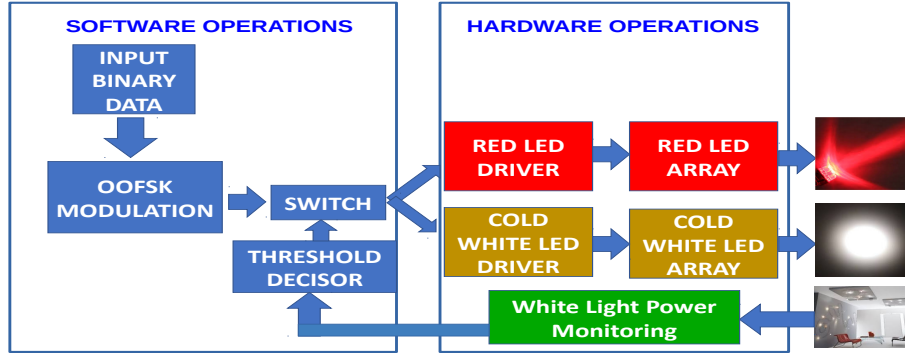


Fig. 2. Main operation allowing color selection in the transmitting path

The sensing stage allowing environmental monitoring in the transmitting stage is essentially a simplified version of the receiver stage, identical from the hardware point of view, but needing a simpler software elaboration. A simple decision-making based on a fuzzy logic has been implemented, by considering a threshold value at the transmitting side, able to activate the transmission of red signals on the basis of the external conditions. Two different LED drivers have been used for the red front-end and the warm white front-end, while the switching operation is directly performed by software, enabling and disabling the proper front-end before and after data transmission. A flow chart describing the proposed algorithm is provided in Figure 3. In this way, when the environmental noise level is too high for allowing a transmission in the cold white spectrum, the red modality is automatically activated until the high level interfering signals will affect the VLC channel. A proper light intensity threshold can be set in the initialization of the algorithm, depending on the average illumination conditions of the operating scenario, while the frequency of light monitoring may be set depending on how changeable the scenario is expected to be. In order to maintain

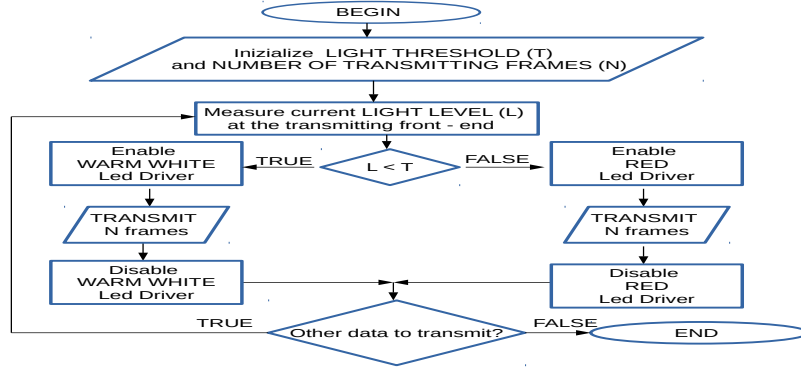


Fig. 3. Flow Diagram of the proposed Dual Color Switching Algorithm

a simple control of this aspect, to avoiding complicate synchronization patterns, light monitoring is performed after N transmitted frames, being N an arbitrary value set in the initialization of the algorithm, as well as the light intensity threshold. Since the switching operation is fully controlled by software, a training mechanism, allowing an intelligent choice of these parameters, could be provided in a future extension of this work. In the present implementation, the receiving front end is equipped with a single photodiode receiving in both LED spectra, while a received noise power threshold is empirically chosen in order to avoid the interference of external lamps in the environment where the system is set up. A further implementation could be performed by replacing the unique wideband photodiode with two distinct narrowband ones, switching on the receiving path with the same thresholds used in the transmission path. Since the signal is created and processed via software, other different adaptive implementations could be performed in a similar way in both transmitting and receiving paths, with few hardware updating. In our system, we performed and tested a non-coherent detection, so the phase of the received signal is arbitrary rather than carefully controlled. In this way, we will be able to guarantee a specific error rate vs SNR regardless the signal phase. This is a very important aspect when low cost components (as shown in Figure 1) are used and a strict phase control is hardly performable as well as a correct timing of all the operations in the receiving phase, in order to obtain a real time demodulation. The On OFF Frequency Shift Keying signal is generated via Arduino Uno, using a dual layer transmitting optical front-end, which is composed of two low power led arrays operating in warm white and red spectra, with a common power supplying, but individually controlled by two identical driving circuit 2. An extremely low-cost optical front-end, composed of a standard photodiode and a trans-impedance amplifier given by cheap operational amplifiers, resistances and capacities, is employed at the receiver side and connected to the analog port of another Arduino Uno (see 5). Signal processing in the receiving stage is performed via software using the

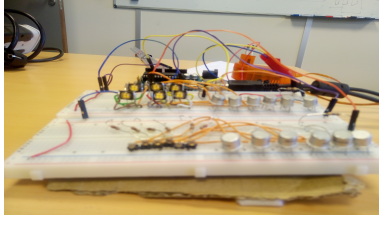


Fig. 4. Transmitting Front-End Prototype

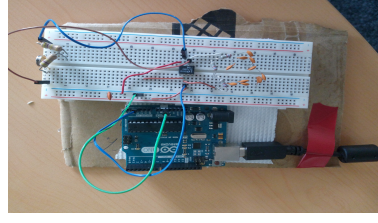


Fig. 5. Receiving Front-End Prototype

commercial software LabView. A lower frequency f_0 is used for codifying 0 and the higher frequency f_1 for codifying 1, as expressed by Eq.1.

$$s(t) = \text{rect}\left(\frac{t}{t_n(t)}\right) \quad (1)$$

where:

$$\begin{cases} t_n(t) = t_0 = \frac{1}{f_0} & \text{if } B_n(t) = 0 \\ t_n(t) = t_1 = \frac{1}{f_1} & \text{if } B_n(t) = 1 \end{cases} \quad (2)$$

and $B_n(t)$ being the signal which represents the binary message to be transmitted. The choice of the modulating signals period with respect to the total bit time t_b is an important aspect in FSK modulation. Choosing two frequencies very far each other allows the demodulator to better distinguish symbols, but, in this case, due to the Nyquist-Shannon theorem, the number of necessary samples for codifying each bit increases, thus resulting in a lower data rate, so a trade off should be achieved considering both the performance architecture of transmitter and receiver, channel conditions and application constraints. The On Off Frequency Shift Keying signal is generated by satisfying the constraints on the modulating frequencies as expressed by Eq.3, with $k=2$ and $m=4$ in order to achieve a good compromise between the number of samples codifying a bit and an easy demodulation of the receiving signal, but respecting a phase continuity of the overall transmitted signal.

$$\begin{cases} f_0 = kf_b = \frac{k}{t_b} \\ f_1 = mf_b = \frac{m}{t_b} \\ m, k \text{ integer} \end{cases} \quad (3)$$

Once the signal is detected, each group of N_b samples, representing the code of a single bit, are collected. The bit time T_b and the sample rate of the receiving stage R_s are related to N_b by Eq.4.

$$N_b = T_b R_s \quad (4)$$

For an easy implementation, an initial value (that is dynamically corrected) for T_b is detected using a known sequence in the transmission path (an alternate

sequence of zero and one), which allows a carrier detection. Furthermore, the group of N_b samples belonged to the q^{th} bit is filtered and processed through an Hann windowing, as defined into Eq.5.

$$h_q(n) = x_q(n)w(n) = x_q(n)\left(\frac{1}{2} - \frac{1}{2}\cos\frac{2\pi n}{N_b}\right) \quad (5)$$

with $0 < n < N_b - 1$ the n^{th} sample codifying the bit, $x_q(n)$ the signal in input to the windowing block, $w(n)$ the Hann function and $h_q(n)$ the signal after the windowing. After this step, a Fast Fourier Transform of the signal is performed, as reported into Eq.6 and Eq.7.

$$H_q = FFT[h_q(n)] = FFT\left[x_q(n)\left(\frac{1}{2} - \frac{1}{2}\cos\frac{2\pi n}{N_b}\right)\right] \quad (6)$$

$$H_q = \sum_{n=0}^{N_b-1} \left[x_q(n)\left(\frac{1}{2} - \frac{1}{2}\cos\frac{2\pi n}{N_b}\right)\right] e^{-j\frac{2\pi kn}{N_b}} \quad (7)$$

Furthermore, the frequency value f_q^* , corresponding to the Fourier Transform peak, is detected and compared to a threshold f_{th} , which is computed as the expected carrier frequency of the transmitting signal.

$$H_q^* = \max[H_q] \quad (8)$$

$$f_q^* = f|_{H(f)=H_q^*} \quad (9)$$

In this way, it is possible to retrieve the original square wave frequencies, thus obtaining the corresponding q^{th} logical level, namely:

$$\begin{cases} B_k = 0 & \text{if } f_q^* < f_{th} \\ B_k = 1 & \text{if } f_q^* \geq f_{th} \end{cases} \quad (10)$$

It is worth to outline that using a low cost hardware for receiving the signal may result in several undesired effects for signal retrieving. In particular, if the frequency clock in both the transmitting and the receiving stage is not stable, it could cause systematic errors in bit retrieving. For this reason, a time frequency graph has been updated for each FFT peak detected in order to rectify the sample rate and overcoming the above problems. All these operations (filtering, windowing, demodulation, etc.), as well as a proper temporization and buffering of data received by the Arduino device, have been performed by properly programming each custom block and developing an ad hoc comprehensive LabView Virtual Instrument (VI). Further details are provided in Section III.

3 Experimental results

A graphic interface, showing both time domain and frequency domain of the receiving signal (as well as the intermediate filtered and windowed waveforms) and storing the received bit sequence, is developed by using the commercial software

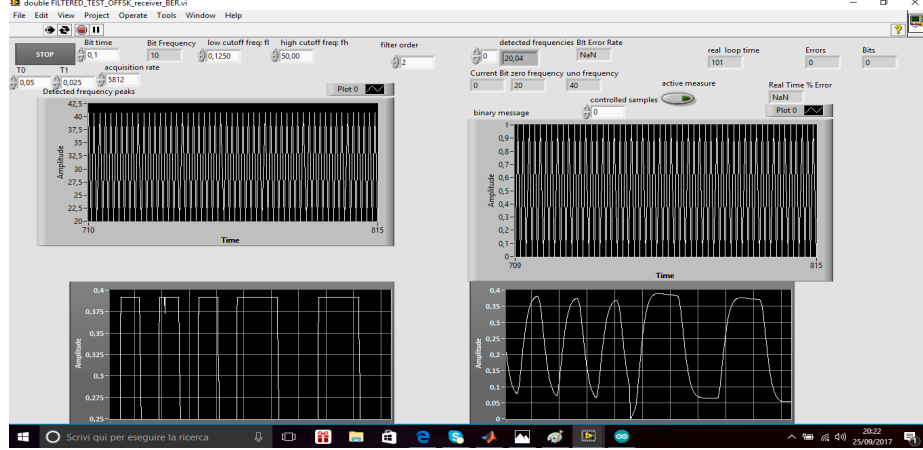


Fig. 6. Graphic User Interface, showing real time original and filtered waveforms, as well as the recovered sequence, during the synchronization stage

Labview. A screen shot, showing the real time signal operations, is provided in Fig. 6, during a synchronization stage, in which an alternate sequence of high and low levels are transmitted and received. As shown in Fig. 6, the correctness of the original time domain waveform (bottom-left in the GUI), the behavior of the Fourier transform peak behavior (top-left), the filtered and windowed signal (bottom-right) and the final retrieved bit sequence (top-right) are assured by a proper softwarization of the signal processing operation and are monitored in real time by the indicators developed in the Graphic User Interface. Furthermore, this interface allows a dynamic change of the system parameters (such as filter position and bandwidth) and the setting of a known sequence in order to achieve a real time evaluation and monitoring of the bit error rate in a desired lapse time (indicators located in the top of the GUI).

In order to experimentally validate the effectiveness of proposed architecture, a series of experiments is set up within two real indoor scenarios, namely an office corridor during normal daily activities, without external lamps turned on (Fig. 7) and a well illuminated hall (Fig. 8)), with several disturbing optical noises. Different communication rates and angles between the transmitter and the receiver are considered in the measurement campaign.

The above two test scenarios have been chosen to properly test the system performance for the following reasons:



Fig. 7. System test bed in low noise conditions

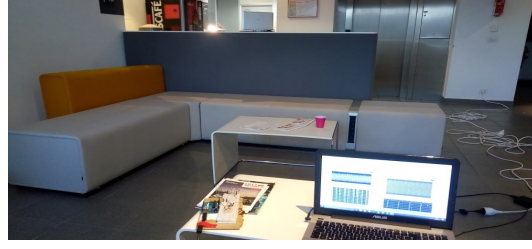


Fig. 8. System test bed in high noise conditions

- Light conditions in Fig. 7 never allow the noisy power to overpass the threshold, so no switch occurs and the warm white modality is always on (a comparison between the two colors are made up by repeating the measurements with the algorithm disabled and feeding only the red front-end in the same conditions).
- Light conditions in Fig. 8 are characterized by a very high noise power, higher than the threshold. In this case, the algorithm enables the red front-end (a comparison between the two colors are made up by repeating the measurements with the algorithm disabled and feeding only the warm front-end in the same condition).

In order to validate the correctness of the proposed dual color approach, the occurred Bit Error Rate (BER) was measured at several distances between transmitter and receiver and each test lasted 100s. Since a maximization of data rates is currently out of the proposal of this work, standard Arduino timing libraries are used in the transmitting stage, allowing a stable communication up to 125bit/s. In Fig. 9, BER measures related to the scenario shown in Fig.7 are shown for both red and warm white modalities.

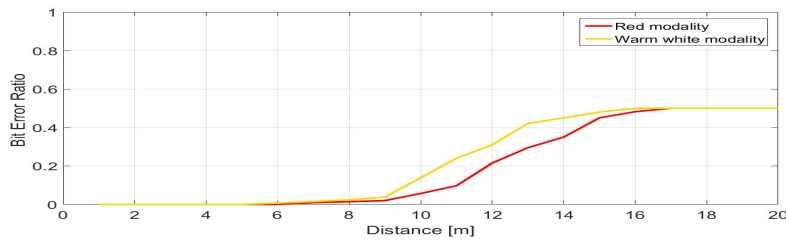


Fig. 9. BER measurements in a low noise environment

Experimental results show that, in a low noise environments, the proposed system properly achieves a considerable distance of 8-9 m in the case of white

cold signals, while maintaining a low BER. In this case, since environmental noise is lower than the empirical threshold, the decision-making process based on fuzzy logic approach does not switch to the red modality, so the warm white led array is effectively used for both lighting and data communication. Identical measurements have been carried out in the scenario shown in Fig. 8, and the obtained BER behavior is shown in Fig. 10.

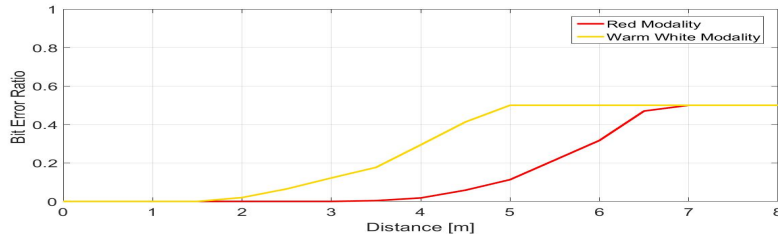


Fig. 10. BER measurements in a high noise environment

In white cold modality, in fact, a degradation of BER can be observed at a distance of 1-2m between the transmitter and the receiver front-ends, so the fuzzy logic algorithm switches to the red modality, allowing to "extend" the communication range by a factor 2. While measurements shown in Fig. 10 demonstrate that, with a proper softwarization, it is possible to achieve good performances in terms of maximum achieved distances even without high power and expensive optical front end, the experimental results shown in Fig. 10 represent a validation on the possibility of using an adaptive multiple color switching in order to dynamically reduce the problem of light condition changes in indoor environments, without preventing the use of the system for both lighting and data communication. Significant limitations of the proposed system are due to the low bandwidth of the Arduino, strongly preventing high rates and the possibility to employ more efficient algorithms like Orthogonal Frequency Division Multiplexing (OFDM) and Color Shift Keying. However, since the proposed approach does not strictly depend neither by the modulation scheme, nor by the employed data acquisition hardware, little modifications could be carried out in order to significantly improve system performance.

4 Conclusion and future perspectives

An adaptive VLC system has been proposed in this work, where a decision-making fuzzy logic-based process allows an opportunistic switching on the best light to transmit, based on the interfering external lights. Validation of this approach has been performed through a test-bed based on Arduino where an On Off Binary Shift Key modulation technique has been implemented. The choice

of this type of modulation is mostly related to the fact that it does not need any type of synchronization. This is fundamental aspect in order to implement a switching approach able to rapidly converge to the best transmitting light selection. In our future perspective, we aim to add more intelligence on the transmitter side, by implementing an Artificial Intelligent (AI) algorithm able to compute the optimal switching value and extend the number and the type of light arrays.

5 Acknowledgements

This work was partially supported by CPER DATA and by 21eme FUI, Store Connect project.

References

1. H. Elgala, R. Mesleh, H. Haas and B. Pricope, "OFDM Visible Light Wireless Communication Based on White LEDs," 2007 IEEE 65th Vehicular Technology Conference - VTC2007-Spring, Dublin, 2007, pp. 2185-2189.
2. L. U. Khan, "Visible Light Communication: applications, architecture, standardization and research challenges," in Elsevier Digital Communications and Networks, vol. 3, pages 78-88, 2017.
3. Q. Wang, D. Giustiniano and D. Puccinelli, "An open source research platform for embedded visible light networking," in IEEE Wireless Communications, vol. 22, no. 2, pp. 94-100, April 2015.
4. Burchardt, H., Serafimovski, N., Tsonev, D., Videv, S., Haas, H.: VLC: Beyond Point-to-Point Communication. In: IEEE Communications Magazine, vol 52, issue 7, pp. 90-105 (2014).
5. Knoblock, F.: Noncoherent Dimming Frequency Shift On-Off Keying Scheme for Low Data Rate Optical Street Lighting Communication. In: ICTION (2015)
6. Ndjongue, A. R., Ferreira, H. C., Ngatched, T. M. N.: Visible Light Communications (VLC) Technology. In: Wiley Encyclopedia of Electrical and Electronics Engineering, pp. 115, (2015)
7. Costanzo S., Di Massa G., Costanzo, A., Borgia, A., Raffo, A. Viggiani, G., Versace, P.: Software-Defined Radar System for Landslides Monitoring. In: New Advances in Information Systems and Technologies. Advances in Intelligent Systems and Computing, vol 445. Springer, Cham (2016)
8. Costanzo, S., Spadafora, F., Di Massa G., Borgia, A., Costanzo, A., Aloï, G., Pace, P., Loscri', V., Moreno, O., H.: Potentialities of usrp-based software defined radar systems . In: Progress In Electromagnetics Research B, Vol. 53, pp. 417435, (2013).